METHOD FOR REDUCING THE SIZE IN DESIGNING A SERVO SYSTEM SIMULATION PROGRAM CONFIGURED IN A PROGRAMMABLE CALCULATOR

BACKGROUND OF THE INVENTION

Field of Invention

[0001]   The present invention generally relates to a method for reducing the size in designing a servo system simulation program, and more particularly, to a method for reducing the size in designing a servo system simulation program configured in a programmable calculator.

Description of Related Art

[0002]   Thanks to the advancements of the technologies, consumer electronic products have also advanced. The continuously growing purposes of the vendors of the consumer electronic products are not only to provide more functions, but also to make these products lighter, thinner, shorter, and smaller in compliance with consumer trends and economic growth.

[0003]   For example, the common programmable calculator (e.g. TI, SHARP and CASIO, ···etc) in use today supports not only +, -, ×, ÷ functions and trigonometric functions, but also supports the input of programmable logic operation equations, can calculate the required mathematic equations, and also has the capability to display a simple coordinate diagram. Addition of a simulation tool to the programmable calculator in the education market would increase its value and stimulate dramatic growth in the education market for programmable calculators.

[0004]   Some difficulties exist in using a conventional programmable calculator as a simulation tool. Using the CASIO fx-6300G as an example, these problems include:

[0005]   1. The difference of the Programming Language: The CASIO fx-6300G calculator only supports 400 steps for use as the standard program steps. An instruction in an assembly language or a high-level language (e.g. BASIC language) may comprise several steps.

[0006]   2. The size of memory is extremely limited: The CASIO fx-6300G only supports 26 registers and it costs another 8 program steps if one extra register is used. Therefore, it is very difficult to use the limited memory of the programmable calculator to configure the simulation of the servo system.

[0007]   3. Reducing the size of the "Simulation Rules" is not easy: The PC automatic control software MATLAB-SIMULINK uses the complicated Runge-Kutta Rule. Although the Runge-Kutta Rule is quite accurate, it uses a lot of memory. Thus, it is very difficult to apply it to the programmable calculator.

[0008]   4. The selection of the servo type is difficult: There is no servo system simulation software that can be used as a reference in the market today, much less for application of the servo system simulation in a programmable calculator. Therefore, it is not easy to apply the input, output and the servo type of the servo system to the programmable calculator.

SUMMARY OF THE INVENTION

[0009]   The present invention provides a method for reducing the size in designing the servo system simulation program configured in a programmable calculator. In order to reduce the memory usage of the programmable calculator, the method redefines the

2

servo types in terms of a general type and redefines the input function as a combination function. Thus, the programmable calculator can be used for servo system simulation.

[0010]   The present invention uses a method to reduce the size in designing the servo system simulation program configured in the programmable calculator. The method of the present invention comprises the steps of: determining the servo type as a general type, and assigning servo parameters; then determining the input function as a combination function, and using a plot to display the input function; the servo type consequently generates an output function and a state variable according to the servo parameters and the input function, and uses a plot to display the output function. The usage of the memory in the programmable calculator is reduced, thus the programmable calculator can be applied to simulate the servo system.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011]   The accompanying drawings are included to provide a further understanding of the invention, and are incorporated in and constitute a part of this specification. The drawings illustrate embodiments of the invention, and together with the description, serve to explain the principles of the invention. In the drawings,

[0012] FIG. 1 is a basic model of the preferred embodiment according to the present invention;

[0013] FIG. 2 is an engineering model of the FIG. 1;

[0014] FIG. 3A is a curve diagram of the step function;

[0015] FIG. 3B is a curve diagram of the ramp function;

[0016] FIG. 3C is a curve diagram of the function, which is a combination of the step function and the ramp function;

[0017] FIG. 4 is a flow chart according to the present invention;

[0018] FIG. 5 is the simulation results according to the present invention;

[0019] FIG. 6A is a basic model of another embodiment according to the present invention;

5   [0020] FIG. 6B is an engineering model of FIG. 6A;

[0021] Table 1 shows the relationship between the servo type and the steady state errors;

[0022] Table 2 shows the planning of the registers of the programmable calculator;

[0023] Table 3 shows the servo system simulation program; and

[0024] Table 4 shows the figures of the simulation results according to the present

10   invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0025]   Fig. 1 shows the basic model of a preferred embodiment according to the present invention. In Fig. 1, the model is represented by the Laplace operator, wherein I

15   represents the input Laplace function, Φ represents the output Laplace function, G represents the gain of the transfer function, A and K represent the pole parameters of the open loop transfer function, H represents the zero parameter of the open loop transfer function, and F represents the feedback value. According to the relationship between the servo type and the steady state errors shown in Table 1, the basic model includes three

20   different servo types, i.e. TYPE 0 servo if K≠0 and A≠0; TYPE I servo if K=0 and A≠0; and TYPE II servo if K=0 and A=0. The formula of the steady state error, resulting from inputting different input functions into these three different servo types, is:

$$\lim_{t \to \infty} \varepsilon(t) = \lim_{s \to 0} s\varepsilon(s)$$

4

Therefore, when the TYPE 0 servo type is applied in the basic model, both of the steady state errors are not equal to 0 if the input function is either a step function or a ramp function. When the TYPE I servo type is applied in the basic model, the steady state error is equal to 0 if the input function is a step function and the steady state error is not equal to 0 if the input function is a ramp function. When the TYPE II servo type is applied in the basic model, both of the steady state errors are equal to 0 if the input function is either a step function or a ramp function.

[0026]   In order to efficiently reduce the memory usage of the programmable calculator (it means using the same CASIO fx-6300G model), the basic model in FIG. 1 uses the Combination Type, a combination of TYPE Φ, TYPE I, and TYPE II, to simulate the servo system. FIG. 2 is an engineering model of FIG. 1 for simulation, wherein the Z(i) and Z(j) are used to represent the state variables.

[0027] The step function and the ramp function are commonly used as an input function in the simulation of the servo system. As the curved diagram of the step function which shown in FIG. 3A, the magnitude of this step function is 0.5. As the curved diagram of the ramp function which shown in FIG. 3B, the initial point of this ramp function is 0, and the slope of this ramp function is 0.5. Δt represents the time interval in both FIG. 3A and FIG. 3B.

[0028]   Similarly, in order to efficiently reduce the memory usage of the programmable calculator, a combination function that is a combination of the step function and the ramp function is used as an input function to simulate the servo system. In FIG. 3C, which shows the curved diagram of the combination function that is a combination of the step function and the ramp function, the initial point of this combination function is 0.5, and the slope is still 0.5.

[0029]   The meaning of each register of the programmable calculator must be defined before the servo simulation system of the programmable calculator starts running. Table 2 shows the planning of the registers of the programmable calculator. As shown in Table 2, 11 programmable calculator registers and 30 non-programmable calculator registers are to be used. Thus, there are 240 (8 x 30) extra steps reduced in the programmable calculator.

[0030]   Table 3 shows the program for simulating the servo system. Before the simulation of the servo system starts running, the engineering model of the servo system is written into the calculator using the given programming language, and uses the registers that planned in Table 2 to store the figures that are calculated out. The program in Table 3 uses 150 instruction steps and 41 registers. In addition to the original 30 extra used registers that use 240 extra steps correspondingly. The total number of the equivalent step used is 390.

[0031]   FIG. 4 shows the flow chart according to the present invention. In order to acquire the largest amount of the memory space to be used in the simulation of the servo system, the contents of the programmable calculator memory is cleared before the simulation of the servo system starts running (S402).

[0032]   The servo type of the servo system is consequently determined. The servo type that uses the engineering model in FIG. 2 as an example uses the engineering model of the general type that is a combination of the TYPE I loop function and the TYPE II loop function. Then, based on this engineering model, the servo parameters as shown in Table 3 are set up: the input function type, the pole parameters, the feedback value and the increase value of the time interval, ...etc. The input function is a combination function

that is a combination of the step function and the ramp function as shown in FIG. 3C (S404).

[0033]   The curve of the input function is then displayed by using a plot after setting the type of the input function (S406). FIG. 5 shows the simulation results, as shown in the diagram, the curve 504 which shown in the screen 502 of the calculator 508 is the curve of the input function.

[0034]   The program of Table 3 is then used to simulate the servo system. The state variables Z[C] and Z[C+16] (that corresponds to the Z(i) and Z(j) as shown in Table 2) are generated from the simulation (S408). After the calculation, the figures of the output function are displayed as a curve diagram of the function (S410). As shown in FIG. 5, the curve 506 shown in the screen 502 of the calculator 508 is the curve of the output function.

[0035]   The number of the executions is increased by one after each time interval. The program in Table 3 simulates the servo system, calculates out the figures of the state variables and the output function, and displays the figures of the output function as a curve diagram (S412). Then the program determines whether the number of executions is less than a predetermined value (the predetermined values used in this embodiment is 15) (S414). If the number of executions is less than the predetermined value, the program recursively returns to step S408, running the step of calculating the figures of the state variables and the output function. If the number of the executions is greater or equal to the predetermined value, the simulation of the servo system is terminated.

[0036]   Table 4 shows the figures of the simulation results according to the present invention. In Table 4, Δt represents the time interval of the servo simulation, Z(i) and Z(j) represent the state variables, wherein Z(j) is also an output function, I/P represents the

input function. One of the simulation results of the engineering model in FIG. 2 is the figures of Z(j), which are displayed as the curve 506 in the screen 502 as shown in the FIG. 5. (The figure Z(j) represented by '*' indicates that the calculator is running out of memory to store the figures of the calculated results.) Therefore, the calculator can be used to simulate the servo system for educational purposes.

[0037]   In all the descriptions above, the more complicated servo system can be simulated when CASIO or other vendors develop a calculator having the larger amount of the memory size. This can be seen from FIG. 6A that shows the basic model of another embodiment according to the present invention, and from FIG. 6B, which shows an engineering model of FIG. 6A. Thus, the calculator can be used to simulate the servo system for educational purposes.

[0038]   The advantages of the present invention are the redefinition of the servo type as a generalized type, redefine the input function as a combination function, to reduce the memory usage of the programmable calculator and thus to apply the calculator using the simulation of the servo system.

[0039]   Although the invention has been described with reference to a particular embodiment thereof, it will be apparent to one of the ordinary skill in the art that modifications to the described embodiment may be made without departing from the spirit of the invention. Accordingly, the scope of the invention will be defined by the attached claims and not by the above detailed description.